Assignment 3: awk

CS3423 - Systems Programming   Rocky Slavin - UTSA

For this assignment, you will use **awk**, **bash**, and some other command-line utilities to create a program for printing user statistics. Your program should take the output from `last` and print the following for each user **with a username matching the abc123 format**:

- Username
- Most recent hostname from which they logged in
- Total time logged in
- Number of times logged in

This assignment requires only awk and bash. **Do not** use sed, Python, or any other languages/utilities.

Example

The example below is an excerpt from the `last` command which your program should be able to take as input. The columns are: username, tty/terminal, host connected from, date logged in, date/time logged out, and total time if the user is not still logged in. Note that, based on whitespace, the number of columns is not consistent. In particular, the date/time logged out column can vary.

You may assume that login and logout time will always be during the same day and users who are still logged in will have logged in no earlier than the current day.

Input:

```
rslavin  pts/2     129.115.27.54    Thu Aug 16 09:31   still logged in
xlg795   pts/1     n0000a061.cs.uts Thu Aug 16 08:31   still logged in
wjq861   pts/1     n0000f64.cs.utsa Thu Aug 16 07:40 - 07:41  (00:01)
michael  pts/1     70.12.32.84      Thu Aug 16 00:18 - 00:25  (00:07)
pwj861   pts/1     71.122.21.84     Wed Aug 15 22:18 - 23:27  (01:08)
daw925   pts/1     cpe-67-11-242-1. Wed Aug 15 21:40 - 21:42  (00:01)
michael  pts/1     79.12.2.4        Wed Aug 15 20:08 - 20:32  (00:24)
hlh735   pts/1     n0000a154.cs.uts Wed Aug 15 17:23 - 17:48  (00:25)
gan122   pts/2     cpe-66-69-13-217 Wed Aug 15 14:00 - 14:04  (00:03)
rkd397   pts/1     n0000a023.cs.uts Wed Aug 15 09:11 - 14:55  (05:43)
iou239   pts/1     123.456.78.9     Wed Aug 15 00:19 - 02:35  (02:16)
iou239   pts/1     17.22.72.102     Tue Aug 14 13:31 - 15:05  (01:34)
xlg795   pts/1     n0000a169.cs.uts Tue Aug 14 16:04 - 17:02  (00:57)
uyi362   pts/15    99-100-170-247.l Tue Aug 14 14:41 - 14:41  (00:00)
js53     pts/7     n0000a023.cs.uts Tue Aug 14 14:27 - 15:06  (00:38)
js73     pts/4     n0000a075.cs.uts Tue Aug 14 14:23 - 15:34  (01:10)
hlh735   pts/1     32.23.32.23.3    Tue Aug 14 12:23 - 13:36  (01:12)
```

Output:

```
User: gan122
        Last host: cpe-66-69-13-217
        Total Time: 0:03
        Total Sessions: 1
User: pwj861
        Last host: 71.122.21.84
        Total Time: 1:08
        Total Sessions: 1
User: hlh735
        Last host: n0000a154.cs.uts
        Total Time: 1:37
        Total Sessions: 2
User: uyi362
        Last host: 99-100-170-247.l
        Total Time: 0:00
        Total Sessions: 1
User: xlg795
        Last host: n0000a061.cs.uts
        Total Time: 15:31
        Total Sessions: 2
User: wjq861
        Last host: n0000f64.cs.utsa
        Total Time: 0:01
        Total Sessions: 1
User: rkd397
        Last host: n0000a023.cs.uts
        Total Time: 5:43
        Total Sessions: 1
User: iou239
        Last host: 123.456.78.9
        Total Time: 5:26
        Total Sessions: 3
User: daw925
        Last host: cpe-67-11-242-1.
        Total Time: 0:01
        Total Sessions: 1
```

Current Time

The current time can be found with the `date` command. In particular, you may need to extract the current hour and/or minute. Use `date --help` to learn how.

IMPORTANT: To make debugging easier on you (and grading easier on me), force `date` to use a specific, static time instead of the actual, current time. Any invocation of `date` *must* be invoked with the following option: `--date='Aug 16 2018 23:05'`. See the following example.

```
$ date --date='Aug 16 2018 23:05'
```

Script Execution

Your program should be invoked through a single bash file (see below) with input taken from stdin. The resulting output should be printed directly to stdout.

```
$ assign3.bash < last.in
```

or

```
$ last | assign3.bash
```

Note that output from `last` may not adhere to the assumptions given so it may be better to use the data provided in `last.in`.

Assignment Data

A sample input file can be found in:
`/usr/local/courses/rslavin/cs3423/Fall18/assign3`.

Script Files

Your program should consist of two files:

- `assign3.bash` - the main file which is initially invoked
- `program.awk` file which is used for an awk invocation run in `assign3.bash`.

Verifying Your Program

You can test your program with the input provided in `last.in` and check that it matches the output above. Your program should also work with arbitrary input from the `last` command. Consider testing your program with `last` on different machines at different times.

Submission

Turn your assignment in via Blackboard. Your zip file, named `LastnameFirstname.zip` should contain only your bash and awk files.